



**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**  
**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO NGHIÊN CỨU KHOA HỌC**

**NHẬN DẠNG DẤU VÂN TAY**

Họ và tên	MSSV
<b>Trần Thanh Tùng</b>	<b>18120258</b>
<b>Nguyễn Viết Dũng</b>	<b>18120167</b>

**Môn : Phương pháp nghiên cứu khoa học**

**Thành phố Hồ Chí Minh - 2021**

---

# Mục lục

---

## Contents

Mục lục.....	2
I. Tóm lược .....	3
II. Giới thiệu.....	4
III. Các nghiên cứu liên quan.....	6
2.1 Nâng cao chất lượng hình ảnh vân tay .....	6
2.1.1 Chuẩn hóa: .....	7
2.1.2 Ước tính ảnh định hướng: .....	7
2.1.3 Ước tính ảnh tần số:.....	8
2.1.4 Bộ lọc Gabor:.....	10
2.2 Bộ phát hiện và mô tả đặc trưng ORB: .....	11
2.2.1 FAST: .....	11
2.2.2 BRIEF: .....	12
IV. Phương pháp đề xuất.....	14
4.1 Nâng cao chất lượng ảnh:.....	14
4.2 Rút trích đặc trưng: .....	15
4.3 So khớp:.....	16
V. Thực nghiệm và kết quả .....	18
5.1 Quy trình thực nghiệm: .....	18
5.2 Kết quả:.....	18
VI. Kết luận .....	20
Tài liệu tham khảo: .....	21

## I. Tóm lược

---

Để phòng việc đánh cắp danh tính, giả mạo thông tin hay sử dụng nhận dạng danh tính để lưu trữ và phân loại. xác minh người sử dụng, lưu trữ danh tính lượng lớn cá thể bằng loại hình đơn giản. Vân tay là đặc điểm sinh trắc học có đặc trưng cá biệt hoàn toàn, trường hợp sinh đôi cũng sẽ có vân tay khác nhau, nhờ đó trở thành công cụ phổ biến nhất để nhận dạng người. Bằng cách sử dụng một số kỹ thuật xử lý ảnh và rút trích đặc trưng nhất định, ta xây dựng một quy trình để xác thực danh tính bằng vân tay. Kết quả sau khi thử nghiệm trên dataset FVC2002 đạt 0.3 EER. Với kết quả như trên thì khi áp dụng trực tiếp, khi xác minh 100 lần sẽ có khoảng 30 lần để lọt kẻ giả mạo/từ chối người dùng thật. Đây là một kết quả ở mức tạm chấp nhận so với những mô hình trước đó, tuy nhiên đây là do việc sử dụng kỹ thuật rút trích ở mức sơ khai, giới hạn ở bộ dataset thử nghiệm, hướng đi này có tiềm năng để phát triển hơn ở tương lai.

## II. Giới thiệu

Trong thời đại 4.0, việc nhận dạng càng ngày càng trở nên quan trọng bởi sự tinh vi trong các hành vi phạm tội. Đánh cắp nhận dạng, giả mạo thông tin đang trở thành vấn nạn trong thời đại thông tin. Trải qua nhiều phương pháp nhận dạng khác nhau như giọng nói, khuôn mặt, vân tay, .. thì vân tay vẫn luôn là giải pháp bảo mật phổ biến, do tính đặc trưng của nó. Dấu vân tay của mỗi người là độc nhất, ngay cả trường hợp anh chị em sinh đôi, vẫn tuân theo quy tắc này.

Việc nhận dạng qua vân tay vẫn luôn được sử dụng và ngày càng cải thiện tính chính xác của nó, hai ứng dụng chính là xác minh (kiểm tra đúng sai của mẫu thử) và định danh (tìm kiếm mẫu thử trong dữ liệu lưu trữ). Trong bài nghiên cứu này, nhóm tập trung cho bài toán xác minh tức là xác định danh tính, ngữ điệu hóa theo ngôn ngữ thường ngày là “liệu bạn có phải là bạn hay không”, thường gặp trong các hệ thống bảo mật bằng vân tay trên laptop, ổ khóa,...

Trong bài toán xác minh danh tính, input là dấu vân tay mà ta cần kiểm chứng, output sẽ cho ra kết quả xác thực xem input này có phải là cùng một dấu vân tay với các dấu vân tay có sẵn trong database hay không (database lưu nhiều ảnh của cùng một dấu vân tay). Ở đây nhóm muốn mô phỏng lại quá trình xác minh danh tính trên các hệ thống bảo mật bằng vân tay ở laptop, ổ khóa,...

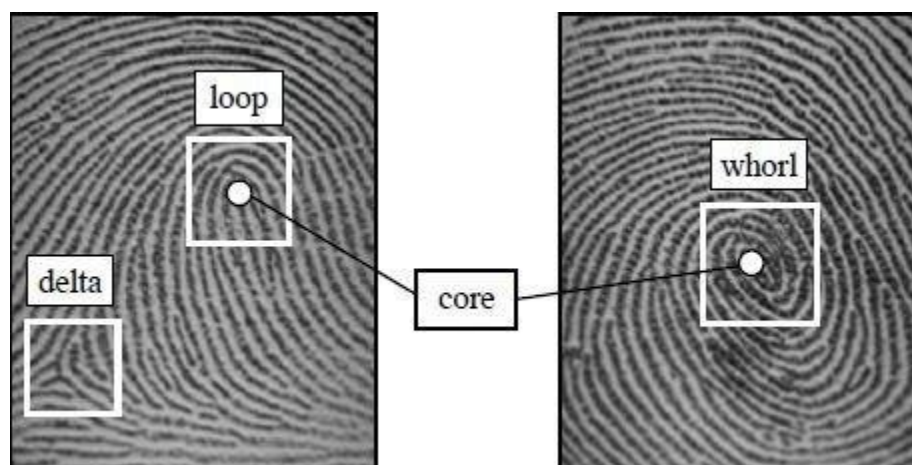
Các cách tiếp cận truyền thống thường tiếp cận vấn đề bằng cách dùng các đặc trưng dựa trên các đặc điểm sinh học của dấu vân tay để so khớp bao gồm:

- Các đường vân: gồm vân gờ và vân lõm.



Hình 1. Vân gờ (Ridge) vân lõm (valley)

- Các loại vân gờ có hình dạng đặc biệt: loop, delta, whorl.



Hình 2. Các loại vân gờ có hình dạng đặc biệt

- Các đường vân gờ bị gãy, không liên tục,... hay còn được gọi là minutiate.

	Termination
	Bifurcation
	Lake
	Independent ridge
	Point or island
	Spur
	Crossover



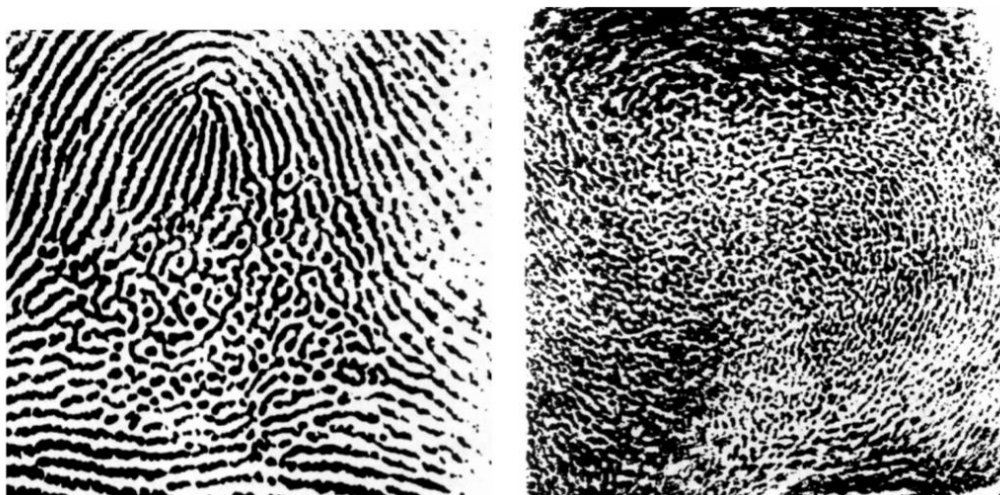
Hình 3. Các loại minutiate

Quy trình được đề xuất trong bài viết có cách tiếp cận khác so với các phương pháp truyền thống vốn dựa trên rút trích minutiae của dấu vân tay, đây là dùng bộ rút trích đặc trưng ORB, không tập trung vào những đặc điểm sinh học của vân tay mà tập trung vào các điểm nổi bật của ảnh. Tuy kết quả ban đầu không thật sự khả quan khi áp dụng cho các hệ thống thực tế nhưng mở ra một hướng mới cho việc nghiên cứu bài toán nhận dạng dấu vân tay.

### III. Các nghiên cứu liên quan

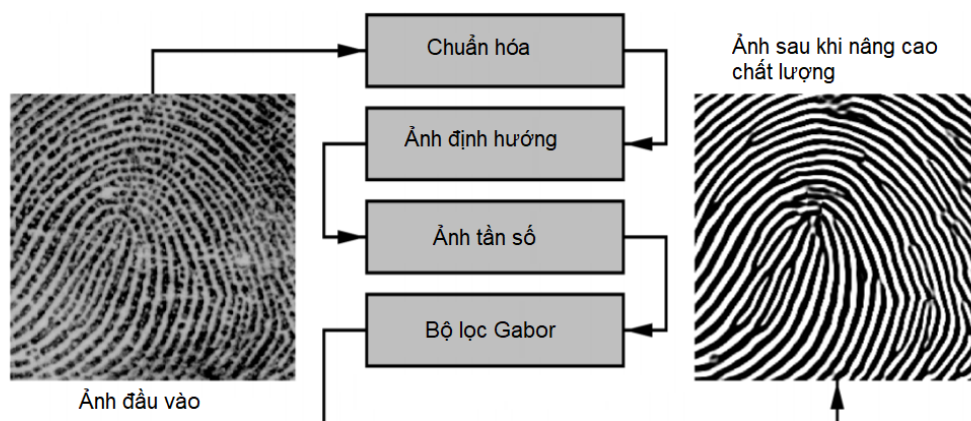
#### 2.1 Nâng cao chất lượng hình ảnh vân tay

[1] Trước bước rút trích đặc trưng của dấu vân tay, người ta thường gặp phải vấn đề lớn khi chất lượng ảnh input đầu vào kém như hình dưới. Chính vì lý do đó, những kỹ thuật nhằm nâng cao chất lượng của ảnh dấu vân tay được phát triển.



Hình 4. Ảnh dấu vân tay có chất lượng kém

Lin Hong và cộng sự đã đề xuất thuật toán để nâng cao chất lượng ảnh dấu vân tay đầu vào như sau:



Hình 5. Flowchart thuật toán nâng cao chất lượng ảnh vân tay đầu vào

Ta sẽ lần lượt đi vào cụ thể từng bước.

**2.1.1 Chuẩn hóa:**

Chuẩn hóa được sử dụng để chuẩn hóa các giá trị cường độ trong hình ảnh bằng cách điều chỉnh mức độ các giá trị điểm ảnh xám để chúng mở rộng trong phạm vi giá trị mong muốn và cải thiện độ tương phản của hình ảnh. Mục tiêu chính của quá trình chuẩn hóa là giảm phương sai của giá trị mức xám dọc theo gờ để tạo điều kiện thuận lợi cho các bước xử lý tiếp theo.

Xác định giá trị cường độ mới của mỗi pixel trong một ảnh bằng:

$$I'[x, y] = \begin{cases} m_0 + \sqrt{(I[x, y] - m)^2 \cdot v_0/v} & \text{Nếu } I[x, y] > m \\ m_0 - \sqrt{(I[x, y] - m)^2 \cdot v_0/v} & \text{không thì} \end{cases}$$

Với  $m$  và  $v$  trung bình và độ lệch chuẩn của ảnh gốc,  $m_0$  và  $v_0$  là trung bình và độ lệch chuẩn của ảnh sau khi chuẩn hóa.

**2.1.2 Ước tính ảnh định hướng:**

Ảnh định hướng đại diện cho thuộc tính nội tại của hình ảnh dấu vân tay và xác định tọa độ bất biến cho các vân gờ và vân lõm trong vùng lân cận địa phương. Các bước chính như sau:

Giả sử  $G$  là ảnh sau khi chuẩn hóa.

- 1) Chia  $G$  thành các block với mỗi block có size  $w \times w$  ( $16 \times 16$ ).
- 2) Tính gradient  $\partial_x(i, j)$  và  $\partial_y(i, j)$  cho mỗi pixel  $(i, j)$
- 3) Ước tính hướng cục bộ của mỗi block có tâm tại pixel  $(i, j)$  bằng cách sử dụng các phương trình sau:

$$V_x(i, j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} 2 \partial_x(u, v) \partial_y(u, v)$$

$$V_y(i, j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} (\partial_x^2(u, v) \partial_y^2(u, v))$$

$$\theta(i, j) = \frac{1}{2} \tan^{-1} \left( \frac{V_y(i, j)}{V_x(i, j)} \right)$$



Với  $\theta(i, j)$  là ước tính bình phương nhỏ nhất của hướng sườn cục bộ tại block có tâm ở pixel  $(i, j)$ .

- 4) Do sự hiện diện của nhiễu, cấu trúc vân gờ và vân lõm bị hỏng,... trong hình ảnh đầu vào, định hướng vân gờ cục bộ  $\theta(i, j)$  có thể không phải lúc nào cũng chính xác. Vì hướng vân gờ cục bộ thay đổi chậm trong vùng lân cận địa phương nơi không có điểm kỳ dị nào xuất hiện, nên bộ lọc low-pass có thể được sử dụng để sửa đổi hướng rằng cục bộ không chính xác. Để thực hiện lọc low-pass, hình ảnh định hướng cần được chuyển đổi thành trường vectơ liên tục, được định nghĩa như sau:

$$\Phi_x(i, j) = \cos(2\theta(i, j))$$

$$\Phi_y(i, j) = \sin(2\theta(i, j))$$

Với  $\Phi_x$  và  $\Phi_y$  lần lượt là 2 thành phần của trường vectơ. Với trường vectơ kết quả, quá trình lọc low-pass có thể được thực hiện như sau:

$$\Phi'_x(i, j) = \sum_{u=-w_\Phi/2}^{w_\Phi/2} \sum_{v=-w_\Phi/2}^{w_\Phi/2} W(u, v) \Phi_x(i - uw, j - vw)$$

và

$$\Phi'_y(i, j) = \sum_{u=-w_\Phi/2}^{w_\Phi/2} \sum_{v=-w_\Phi/2}^{w_\Phi/2} W(u, v) \Phi_y(i - uw, j - vw)$$

$W$  là bộ lọc low-pass 2 chiều  $w_\Phi \times w_\Phi$  chỉ định kích thước của bộ lọc. Lưu ý rằng hoạt động làm mịn được thực hiện ở cấp block. Kích thước mặc định của bộ lọc là  $5 \times 5$ .

- 5) Tính toán hướng vân gờ cục bộ tại  $(i, j)$  bằng cách:

$$O(i, j) = \frac{1}{2} \tan^{-1} \left( \frac{\Phi'_y(i, j)}{\Phi'_x(i, j)} \right)$$

### 2.1.3 Ước tính ảnh tần số:

Trong một vùng lân cận địa phương nơi không có minutiae và điểm kỳ dị nào xuất hiện, các mức xám dọc theo các vân gờ và vân lõm có thể được mô hình hóa dưới dạng sóng hình sin dọc theo hướng của vân gờ đó. Do đó, tần số vân gờ cục bộ là một đặc tính nội tại khác của ảnh dấu vân tay. Gọi  $G$  là ảnh chuẩn hóa và  $O$  là ảnh định hướng, sau đó các bước liên quan đến ước tính tần số sườn núi cục bộ như sau:

- 1) Chia  $G$  thành các block với mỗi block có size  $w \times w$  ( $16 \times 16$ ).
- 2) Với mỗi block có tâm tại pixel  $(i, j)$ , tính một cửa sổ định hướng có kích thước là  $l \times w$  ( $32 \times 16$ )
- 3) Với mỗi block có tâm tại pixel  $(i, j)$ , tính x-signature,  $X[0], X[1], \dots, X[l-1]$  của vân gờ và vân lõm nằm trong cửa sổ định hướng với:

$$X[k] = \frac{1}{w} \sum_{d=0}^{w-1} G(u, v), k = 0, 1, \dots, l-1$$



$$u = i + \left(d - \frac{w}{2}\right) \cos O(i, j) + \left(k - \frac{l}{2}\right) \sin O(i, j),$$

$$v = j + \left(d - \frac{w}{2}\right) \sin O(i, j) + \left(\frac{l}{2} - k\right) \cos O(i, j).$$

Nếu không có minutiae và điểm kỳ dị nào xuất hiện trong cửa sổ định hướng, thì x-signature tạo thành một sóng hình sin rời rạc, có cùng tần số với tần số của các vân gờ và vân lõm trong cửa sổ định hướng. Do đó, tần số của các vân gờ và vân lõm có thể được ước tính từ x-signature. Gọi  $\mathcal{T}(i, j)$  là số pixel trung bình giữa hai đỉnh liên tiếp trong x-signature, sau đó tần số  $\Omega(i, j)$  được tính là:  $\Omega(i, j) = 1/\mathcal{T}(i, j)$ . Nếu không có đỉnh liên tiếp nào có thể được phát hiện từ x-signature, thì tần số được gán giá trị -1 để phân biệt với các giá trị tần số hợp lệ.

- 4) Đối với hình ảnh dấu vân tay được quét ở độ phân giải cố định, giá trị tần suất của các vân gờ và vân lõm trong vùng lân cận địa phương nằm trong một phạm vi nhất định. Đối với hình ảnh 500dpi, khoảng này là  $[1/3, 1/25]$ . Do đó, nếu giá trị ước tính của tần số nằm ngoài phạm vi này, thì tần số được gán giá trị -1 để chỉ ra rằng không thể thu được tần số hợp lệ.
- 5) Các block trong đó các minutiae hoặc các điểm kỳ dị xuất hiện hay các vân gờ và vân lõm bị hỏng không tạo thành một sóng hình sin rõ ràng. Các giá trị tần số cho các khối này cần được nội suy từ tần số của các khối lân cận có tần số xác định rõ. Nội suy được thực hiện như sau:

i. Với mỗi block có tâm tại pixel  $(i, j)$

$$\Omega'(i, j) = \begin{cases} \Omega(i, j) & \text{Nếu } \Omega(i, j) \neq -1 \\ \frac{\sum_{u=-w_\Omega/2}^{w_\Omega/2} \sum_{v=-w_\Omega/2}^{w_\Omega/2} w_g(u, v) \mu(\Omega(i - uw, j - vw))}{\sum_{u=-w_\Omega/2}^{w_\Omega/2} \sum_{v=-w_\Omega/2}^{w_\Omega/2} W_g(u, v) \delta(\Omega(i - uw, j - vw) + 1)} \end{cases}$$

Với

$$\mu(x) = \begin{cases} 0 & \text{Nếu } x \leq 0 \\ x & \text{Không thì} \end{cases}$$

$$\delta(x) = \begin{cases} 0 & \text{Nếu } x \leq 0 \\ 1 & \text{Không thì} \end{cases}$$

$W_g$  là một Gaussian kernel rời rạc với giá trị trung bình và phương sai lần lượt là 0 và 9, và  $w_\Omega = 7$  là kích thước của kernel.

- ii. Nếu tồn tại ít nhất 1 block có giá trị -1, thì đổi trị của  $\Omega$  và  $\Omega'$  cho nhau và quay lại bước i.

#### 2.1.4 Bộ lọc Gabor:

Bộ lọc Gabor có cả đặc tính chọn lọc tần số và chọn lọc định hướng và có độ phân giải khớp tối ưu trong cả miền không gian và miền tần số. Do đó, bộ lọc Gabor thích hợp để loại bỏ nhiễu và bảo tồn cấu trúc vân gờ/vân lõm thực sự.

Bộ lọc Gabor có cấu trúc:

$$h(x, y; \phi, f) = \exp \left\{ -\frac{1}{2} \left[ \frac{x_\phi^2}{\delta_x^2} + \frac{y_\phi^2}{\delta_y^2} \right] \right\} \cos(2\pi f x_\phi),$$

$$x_\phi = x \cos \phi + y \sin \phi,$$

$$y_\phi = -x \sin \phi + y \cos \phi$$

trong đó  $\phi$  là hướng của bộ lọc Gabor,  $f$  là tần số của sóng phẳng hình sin, và  $\delta_x$  và  $\delta_y$  là các hằng số không gian của đường bao Gaussian dọc theo các trục  $x$  và  $y$ , tương ứng. Chức năng truyền điều chế (modulation transfer function) của bộ lọc Gabor có thể được biểu diễn dưới dạng:

$$h(x, y; \phi, f) = 2\pi\delta_x\delta_y \exp \left\{ -\frac{1}{2} \left[ \frac{(u_\phi - u_0)^2}{\delta_u^2} + \frac{(v_\phi - v_0)^2}{\delta_v^2} \right] \right\} +$$

$$2\pi\delta_x\delta_y \exp \left\{ -\frac{1}{2} \left[ \frac{(u_\phi + u_0)^2}{\delta_u^2} + \frac{(v_\phi + v_0)^2}{\delta_v^2} \right] \right\},$$

$$u_\phi = u \cos \phi + v \sin \phi$$

$$v_\phi = -u \sin \phi + v \cos \phi,$$

$$u_0 = \frac{2\pi \cos \phi}{f},$$

$$v_0 = \frac{2\pi \sin \phi}{f}$$

Với  $\delta_u = 1/2\pi\delta_x$ ,  $\delta_v = 1/2\pi\delta_y$

Để áp dụng bộ lọc Gabor cho một ảnh, ba tham số phải được chỉ định:

- tần số của sóng phẳng hình sin,  $f$
- hướng bộ lọc
- độ lệch chuẩn của đường bao Gauss,  $\delta_x$  và  $\delta_y$ .

Rõ ràng, đặc tính tần số của bộ lọc,  $f$ , hoàn toàn được xác định bởi tần số đỉnh cục bộ và định hướng được xác định bởi định hướng vân gờ cục

bộ. Việc lựa chọn các giá trị của  $\delta_x$  và  $\delta_y$  liên quan đến sự cân bằng. Các giá trị càng lớn, bộ lọc càng mạnh để nhiễu nhưng bộ lọc càng có nhiều khả năng tạo ra các gờ và thung lũng giả. Mặt khác, các giá trị  $\delta_x$  và  $\delta_y$  càng nhỏ thì khả năng các bộ lọc sẽ tạo ra các gờ và thung lũng giả càng ít; do đó kém hiệu quả hơn trong việc loại bỏ nhiễu. Các giá trị của  $\delta_x$  và  $\delta_y$  lần lượt được đặt thành 4,0 và 4,0 dựa trên dữ liệu thực nghiệm. Gọi  $G$  là ảnh dấu vân tay chuẩn hóa,  $O$  là ảnh định hướng, là  $\mathcal{F}$  hình ảnh tần số, hình ảnh nâng cao  $\mathcal{E}$  thu được như sau:

$$\mathcal{E}(i, j) = \left\{ \sum_{u=-w_g/2}^{w_g/2} \sum_{v=-w_g/2}^{w_g/2} h(u, v: O(i, j), \mathcal{F}(i, j)) G(i - u, j - v) \right.$$

Với  $w_g = 11$  là kích thước của bộ lọc Gabor.

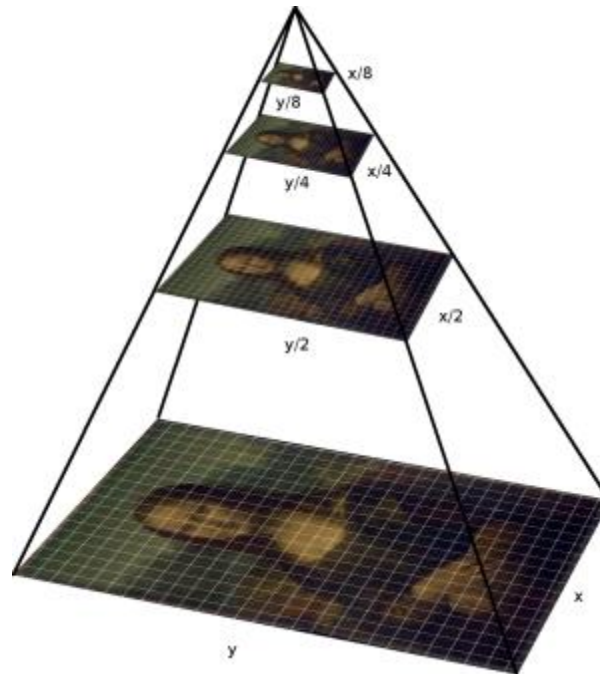
## 2.2 Bộ phát hiện và mô tả đặc trưng ORB:

[2] ORB là một thuật toán tạo phát hiện và mô tả đặc trưng ra bởi Ethan Rublee và cộng sự nhằm thay thế cho SIFT và SURF (được đăng ký bằng sáng chế → phải mất phí khi sử dụng).

Tổng quan, ORB được kết hợp từ FAST( Features from accelerated segment test) tìm kiếm điểm đặc trưng và BRIEF (Binary Robust Independent Elementary Feature ) mô tả hình ảnh với chỉnh sửa để cải thiện hiệu suất xử lý, nhà phát triển đã có sửa đổi để cải thiện hiệu suất, giúp bất biến trước phép quay.

### 2.2.1 FAST:

Đầu tiên, ORB sử dụng FAST để phát hiện các điểm đặc trưng và sử dụng phương pháp đo điểm góc Harris để chọn N điểm đặc trưng có giá trị phản hồi góc Harris lớn nhất từ các điểm đặc trưng FAST. Song song lúc đó, nó cũng tạo ra các kim tự tháp Gaussian, để có thể bất biến với phép thu phóng.



Hình 6. Kim tự tháp Gaussian [3]

Tuy nhiên, FAST không tính toán theo định hướng, điều này khiến FAST không bất biến với phép quay. Các tác giả đã giải quyết vấn đề này một cách đầy khéo léo như sau:

Đầu tiên tính trọng số cường độ của patch với góc nằm ở tâm ảnh. Hướng của vector từ góc chỉ đến tâm cho ta định hướng. Để cải thiện tính bất biến với phép quay, trọng số cường độ được tính với cả  $x$  và  $y$  nằm trong vùng bán kính của patch.

### 2.2.2 BRIEF:

Bộ mô tả BRIEF là một mô tả thuộc tính mới (bộ mô tả điểm đặc trưng) có sử dụng các kiểm tra nhị phân đơn giản giữa các điểm ảnh trong hình ảnh của một bản vá nhẵn. Hiệu quả của nó tương tự như SIFT ở nhiều khía cạnh, trong đó rất tốt với ánh sáng, làm mờ và hướng nhìn méo. Tuy nhiên, nó rất nhạy cảm với phép quay trong.

Vì vậy, nhóm nghiên cứu đã tinh chỉnh thành một biến thể để nó có thể bất biến với phép quay và gọi nó là rBRIEF (rotated BRIEF).

Đầu tiên dựa vào các đặc trưng định hướng (đã được rút trích từ FAST), cho bất kỳ set đặc trưng có kích thước  $n$  nào nằm ở pixel  $(x,y)$ , định nghĩa một ma trận kích thước  $2 \times n$  gọi là  $S$ , chứa tọa độ của các điểm này. Sau đó dùng định hướng patch  $\theta$ , tính được ma trận xoay của  $S$  và gọi là  $S_\theta$ .

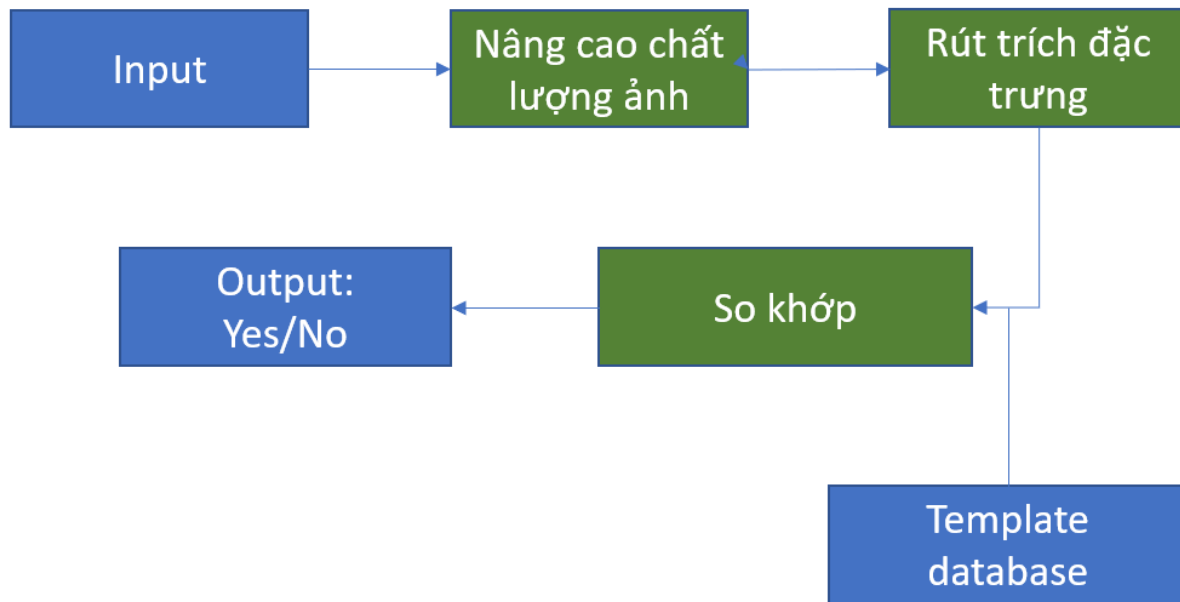
Tiếp đó ORB tùy chỉnh góc quay tăng 12 độ và xây dựng một bảng tra cứu của các pattern BRIEF đã được tính toán. Miễn là yếu điểm định hướng  $\theta$  là cố định thì số lượng các điểm  $S_\theta$  sẽ được dùng để được tính toán bộ mô tả của nó.

BRIEF có một đặc điểm quan trọng là mỗi điểm quan trọng có phương sai lớn và giá trị trung bình gần 0,5. Nhưng một khi các yếu điểm đã được tính định hướng, nó mất đặc điểm này và trở nên phân phối hơn. Phương sai cao làm cho đặc trưng dễ phân biệt hơn, vì nó phản ứng khác biệt với các input. Một tính chất mong muốn khác là không có các thử nghiệm không tương quan với nhau, vì vậy mỗi thử nghiệm sẽ đóng góp vào kết quả cuối cùng. Để giải quyết các vấn đề này, ORB chạy một giải thuật tìm kiếm tham lam khắp các thử nghiệm để tìm được trường hợp mà có phương sai lớn và giá trị trung bình gần 0.5.

## IV. Phương pháp đề xuất

Mô hình nhóm đưa ra dựa theo mô hình đề xuất của Ivona Najdenkoska [4].

Hình dưới đây thể hiện cụ thể các bước thực hiện quá trình xác minh được đề xuất:



Hình 7. Mô hình xác minh đề xuất

Ở đây ta cố gắng mô phỏng lại một hệ thống xác minh danh tính bằng dấu vân tay hay gập trên laptop hoặc ổ khóa. Các thành phần của mô hình:

**Input:** là ảnh một dấu vân tay bất kỳ cần xác minh.

**Nâng cao chất lượng ảnh:** ảnh dấu vân tay cần xác minh sẽ được nâng cao chất lượng, làm tiền đề cho bước rút trích đặc trưng.

**Rút trích đặc trưng:** từ ảnh đã được nâng cao chất lượng, tiến hành rút trích đặc trưng của nó để chuẩn bị so khớp.

**Template database:** chứa các đặc trưng của các ảnh thể hiện cho cùng 1 dấu vân tay gốc (tức là dấu vân tay của người dùng thật).

**So khớp:** So khớp đặc trưng của ảnh cần được xác minh với các đặc trưng của ảnh trong template database để xác định kết quả trả về.

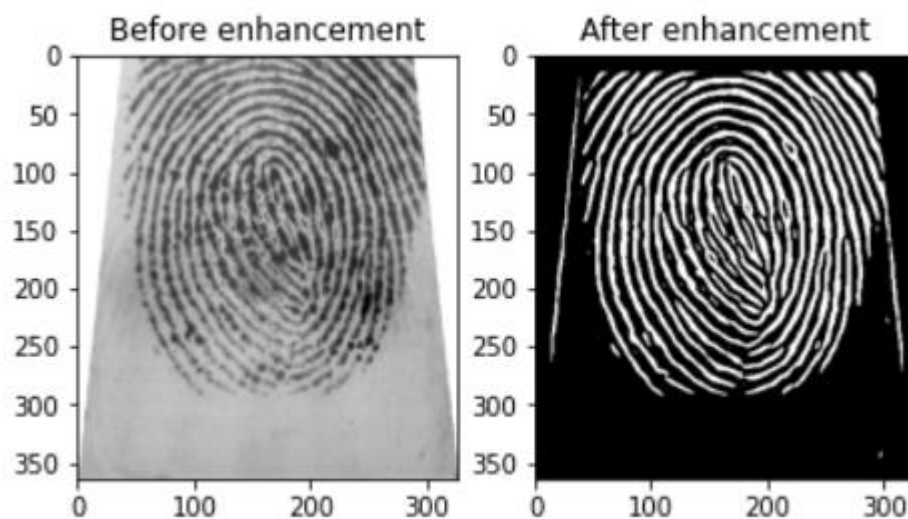
**Output:** Yes (dấu vân tay đó cũng là dấu vân tay gốc trong template database), No (dấu vân tay đó không phải là dấu vân tay gốc trong template database)

### 4.1 Nâng cao chất lượng ảnh:

Chất lượng ảnh input đầu vào có ảnh hưởng rất lớn đến hiệu suất của hệ thống, một số trường hợp có khiến cho ảnh có chất lượng rất kém vì nhiều nguyên nhân chủ quan và khách quan: điều kiện da tay, bề mặt cảm biến,

áp lực tiếp xúc giữa vân tay và cảm biến quá mạnh hoặc quá yếu, chất lượng cảm biến,...

Phương pháp được sử dụng ở đây là phương pháp của Lin Hong và cộng sự được đề cập ở trên cộng với quá trình tách vùng dấu vân tay. Phương pháp tách vùng dấu vân tay được Peter Kovesi [5] đề xuất như sau: Sau khi đã chuẩn hóa, ảnh sẽ được chia thành các block với kích thước  $16 \times 16$ . Xét mỗi block, tính độ lệch chuẩn của nó, nếu lớn hơn threshold cho sẵn (0.1) thì nó sẽ được coi là một phần của dấu vân tay. Kết quả của quá trình giống như hình ảnh dưới đây:



Hình 8. Ảnh dấu vân tay trước và sau khi nâng cao chất lượng

## 4.2 Rút trích đặc trưng:

Sử dụng bộ rút trích đặc trưng ORB. Về cơ bản ORB là một kết hợp giữa bộ phát hiện yếu điểm FAST và bộ mô tả BRIEF với nhiều sửa đổi để cải thiện hiệu suất.

Ta cấu hình ORB như sau:

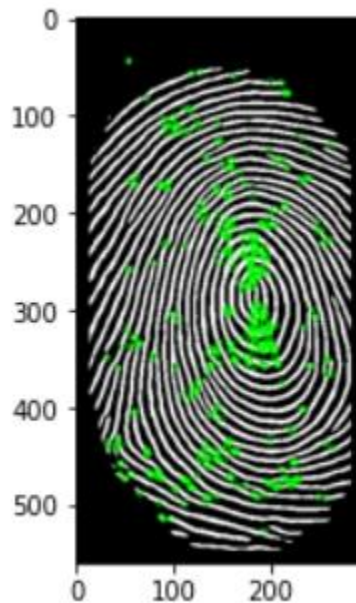
- Nfeatures = 500: là số đặc trưng tối đa của ảnh được giữ lại sau khi thực hiện thuật toán.
- scaleFactor = 1.2: Tỷ lệ phân cấp của kim tự tháp, nghĩa là cấp sau có số lượng pixel nhỏ hơn cấp trước đó  $1.2 \times 1.2$  lần
- nlevels = 8: Số lượng cấp độ sử dụng trong kim tự tháp Gaussian.
- edgeThreshold = 31: kích thước của đường biên khi các đặc trưng không được phát hiện.
- firstLevel = 0: Cấp độ khi bỏ ảnh vào kim tự tháp.
- WTA\_K = 2: Số lượng điểm để tạo ra một phần tử của bộ mô tả BRIEF định hướng. Giá trị mặc định 2 có nghĩa là BRIEF lấy một cặp điểm ngẫu nhiên và so sánh độ sáng của chúng, nhận về kết quả 0/1.



- `scoreType = ORB::HARRIS_SCORE`: có nghĩa là thuật toán Harris được sử dụng để xếp hạng các đặc trưng.
- `patchSize = 31`: kích thước patch dùng bởi BRIEF.
- `fastThreshold = 20`: ngưỡng fast.

Cách hoạt động của ORB được mô tả ngắn gọn như sau: ORB phát hiện các yếu điểm của ảnh, sau đó dùng các yếu điểm này để trả về bộ mô tả của ảnh đó.

Kết quả các yếu điểm sau khi rút trích:



Hình 9. Kết quả các yếu điểm sau khi rút trích

### 4.3 So khớp:

Khi so khớp 2 ảnh, tức là đang so khớp 2 bộ mô tả của 2 ảnh đó được thu về sau quá trình rút trích đặc trưng. Sử dụng bộ phương pháp brute force matching, tức là đối sánh vét cạn. Hiểu đơn giản, nó lấy đặc trưng đầu tiên bộ thứ nhất và tiến hành so khớp với tất cả đặc trưng trong bộ thứ hai bằng cách sử dụng một số tính toán khoảng cách. 2 đặc trưng giống nhau nhất (khoảng cách bé nhất) được trả về.

Ta cấu hình Brute-force matcher như sau:

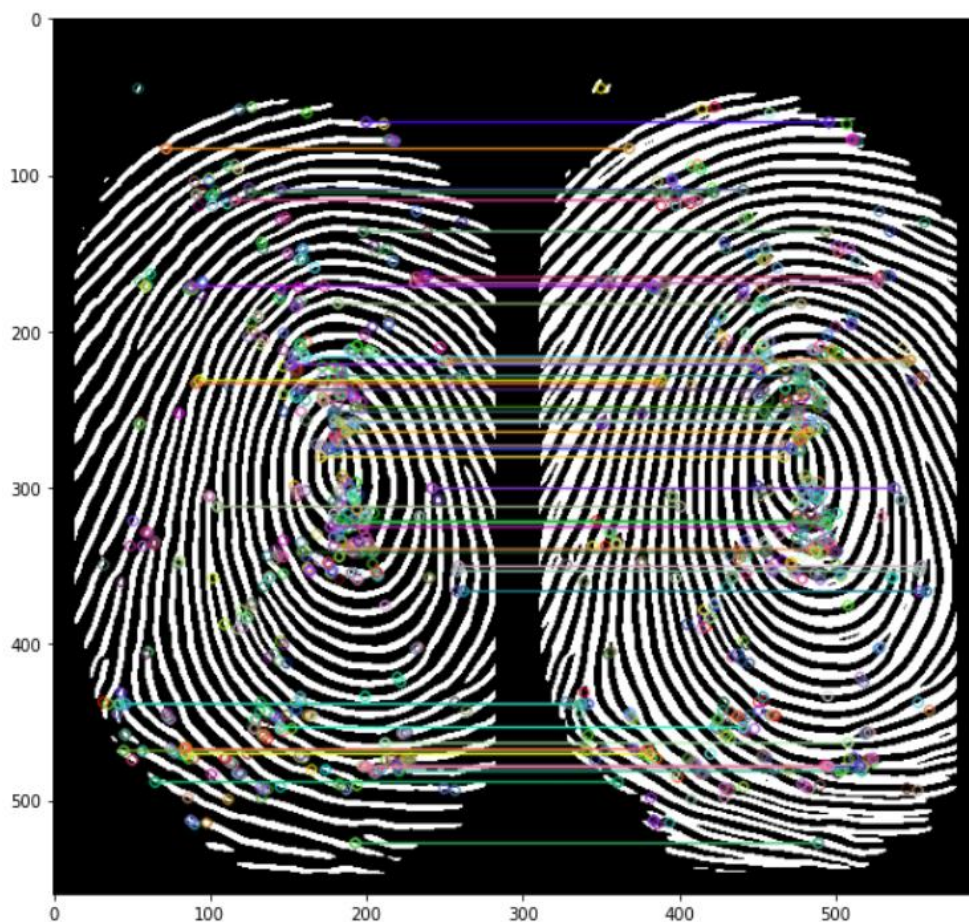
- `normType = NORM_HAMMING`: tức là dùng khoảng cách hamming so khớp.
- `crossCheck = True`: nghĩa là chỉ trả về 1 cặp đặc trưng giống nhau nhất (1:1).

Sau khi so khớp giữa 2 ảnh, sẽ trả về một bộ các cặp đặc trưng được cho là giống nhất. Mỗi cặp có 1 thông số là distance tương trưng cho khoảng cách hamming của cặp đặc trưng đó.

Để tránh lấy phải những cặp đặc trưng rác (không giống nhau) ta chọn một threshold để giới hạn sự khác nhau của các cặp đặc trưng.

Sau đó ta xét tổng số cặp đặc trưng, nếu vượt threshold cho trước thì quyết định rằng 2 dấu vân tay này khớp với nhau và ngược lại.

Thực hiện lần lần cho tất cả các ảnh trong template database, nếu ảnh cần xác minh khớp hơn 1 nửa ảnh trong template database thì output kết quả là Yes còn ngược lại là No.



Hình 10. So khớp các điểm đặc trưng giữa 2 dấu vân tay

## V. Thực nghiệm và kết quả

Link github cho quá trình thực nghiệm : <https://github.com/tungrg/fpr1>

Ta thực nghiệm mô hình trên dataset FVC2002.

Mỗi dataset con có 10 dấu vân tay với 8 ảnh thể hiện của mỗi dấu vân tay đó, nghĩa là mỗi dataset con có 80 dấu vân tay tổng.

FCV2002				
	Loại cảm biến	wxd	Kích thước ảnh	Độ phân giải
DB1_B	Cảm biến quang	10x8	388x374 (142 Kpixels)	500 dpi
DB2_B	Cảm biến quang	10x8	296x560 (162 Kpixels)	569 dpi
DB3_B	Cảm biến điện dung	10x8	300x300 (88 Kpixels)	500 dpi
DB4_B	Dữ liệu tổng hợp được tạo	10x8	288x384 (108 Kpixels)	Khoảng 500 dpi

Thông số để đánh giá mô hình:

EER: điểm cân bằng của FAR (False accept rate – tỷ lệ chấp nhận người dùng giả mạo) và FRR (False reject rate – tỷ lệ từ chối người dùng thật)

### 5.1 Quy trình thực nghiệm:

Với mỗi dataset con, ta chia dataset thành 3 tập là tập train, tập valid và tập test theo tỷ lệ 50%:25%:25%. Cụ thể hơn là với mỗi dấu vân tay có 8 ảnh thể hiện thì tập train có 4 ảnh, tập valid có 2 ảnh, tập test có 2 ảnh.

Nguyên nhân cho quá trình này trong quá trình so khớp, ta cần tìm 2 thông số là threshold giới hạn sự khác nhau của các đặc trưng và threshold để xác định 2 dấu vân tay là khớp với nhau.

Tập train ở đây nên hiểu như là một template database.

Tập valid dùng để tìm các thông số threshold.

Tập test là thể thử nghiệm khách quan xem hiệu quả của mô hình đề xuất.

Cách thực nghiệm như sau:

- 1) Đối với tập train, chia thành các template database theo dấu vân tay.  
Nghĩa là sẽ chia thành 10 template database, với mỗi database chứa 4 ảnh thể hiện của cùng 1 dấu vân tay.  
Tập test và tập valid vẫn giữ nguyên, mỗi tập có 2 ảnh thể hiện của các dấu vân tay, tức 20 ảnh cho mỗi tập.
- 2) Dùng tập valid để so khớp với lần lượt với từng template database đã được tách ra từ tập train để chọn bộ thông số threshold tối ưu của mô hình.
- 3) Dùng tập test để đánh giá khách quan hiệu quả của mô hình.

### 5.2 Kết quả:

Sau khi thực nghiệm ta thu được kết quả như sau:

Thời gian xác minh trung bình: 3.4364 (s).

	DB1_B	DB2_B	DB3_B	DB4_B	Average
EER	0.2833	0.1722	0.2861	0.475	0.3042

*Bảng kết quả EER tổng quát trên bộ dữ liệu*

So sánh với các phương pháp trước đó:

	EER
Phương pháp đề xuất	0.3042
A minutia-based partial fingerprint recognition system (2005) [6]	0.0169
Use of ridge points in partial fingerprint matching (2007) [7]	0.0937

*Bảng so sánh kết quả*

---

## VI. Kết luận

---

Tuy mô hình đề xuất không đạt kết quả thực sự cao trên bộ dữ liệu thử nghiệm, không thể áp dụng vào thực tế ngay được nguyên nhân phần nhiều vì sự hạn chế kinh nghiệm trong việc thiết kế mô hình của nhóm nhưng vẫn là một cách tiếp cận mới cho bài toán nhận dạng vân tay. Trong tương lai vẫn còn nhiều khả năng để phát triển tiếp cho mô hình này như vừa kết hợp các phương pháp rút trích truyền thống với ORB,...

---

## Tài liệu tham khảo:

---

- [1] L. Hong, Y. Wan and A. Jain, "Fingerprint image enhancement: algorithm and performance evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 777 - 789, 1998.
- [2] E. Rublee, V. Rabaud, K. Konolige and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *2011 International Conference on Computer Vision*, Barcelona, 2011.
- [3] "Giới thiệu về ORB (NHANH CHÓNG được định hướng và xoay vòng)," [Trực tuyến]. Available: <https://ichi.pro/vi/gioi-thieu-ve-orb-nhanh-chong-duoc-dinh-huong-va-xoay-vong-72709114183887>. [Đã truy cập 10 6 2021].
- [4] I. Najdenkoska, "Fingerprint recognition," [Online]. Available: <https://github.com/ivonajdenkoska/fingerprint-recognition>. [Accessed 4 4 2021].
- [5] D. P. Kovesi, "Fingerprint-Enhancement-Python," [Online]. Available: <https://github.com/Utkarsh-Deshmukh/Fingerprint-Enhancement-Python>. [Accessed 10 6 2021].
- [6] Tsai-YangJea, "A minutia-based partial fingerprint recognition system," in *Pattern Recognition*, 2005, pp. 1672-1684.
- [7] S. N. S. H. S. a. P. P. G. Fang, "Use of ridge points in partial fingerprint matching," *Biometric Technology for Human Identification IV*, vol. 6539, 2007.